UCLA
COMPUTATIONAL AND APPLIED MATHEMATICS

Efficient Variants of the Vertex Space Domain Decomposition Algorithm

Tony F. Chan
Tarek P. Mathew
Jian-Ping Shao

January 1992

CAM Report 92-07

Department of Mathematics
University of California, Los Angeles
Los Angeles, CA. 90024-1555

EFFICIENT VARIANTS OF THE VERTEX SPACE DOMAIN DECOMPOSITION ALGORITHM

January 28, 1992

TONY F. CHAN , TAREK P. MATHEW AND JIAN-PING SHAO *

Abstract. We describe several variants of the vertex space algorithm of Smith for two dimensional elliptic problems. The vertex space algorithm is a domain decomposition method based on non-overlapping subregions, in which the reduced Schur complement system on the interface is solved using a generalized block Jacobi type preconditioner, with the blocks corresponding to the vertex space, edges and a coarse grid. We consider two kinds of approximations for the edge and vertex space sub-blocks, one based on Fourier approximation, and another based on an algebraic *probing* technique in which sparse approximations to these sub-blocks are computed. Our motivation is to improve efficiency of the algorithm without sacrificing the optimal convergence rate. Numerical and theoretical results on the performance of these algorithms, including variants of an algorithm of Bramble, Pasciak and Schatz are presented.

Key Words. Domain decomposition, schur complement, interface probe, block Jacobi preconditioner, elliptic equations, preconditioners, vertex spaces.

AMS subject classifications: 65N20, 65F10.

1. Introduction. Domain decomposition methods often provide suitable techniques for solving large linear systems of equations arising from discretizations of partial differential equations. In particular, these methods can be advantageous for the efficient and localized treatment of irregular geometries, discontinuous coefficients, local grid refinement, boundary layers and coupling between equations of different type, see for instance [20, 5, 6, 7].

In this paper, we primarily focus on the development of efficient versions of two divide and conquer type domain decomposition algorithms based on non-overlapping subregions for solving self adjoint elliptic problems in two dimensions. The algorithms we describe are variants of the vertex space algorithm (VS) proposed by Smith [27] and Nepomnyaschikh [25], and an algorithm of Bramble, Pasciak and Schatz (BPS) [3]. In both cases, a block Jacobi type preconditioner is used to solve the reduced Schur complement system on the interface. The blocks in the BPS algorithm correspond to the nodes on the edges separating the subdomains and to the collection of vertices of the subregions, while in the vertex space algorithm additional overlapping blocks, centered about each vertex consisting of nodes on the interface close to the vertex, are included to account for coupling amongst the non-overlapping blocks.

In order to implement the original version of the VS preconditioner [27], the sub-blocks of the Schur complement, which are dense matrices, need to be computed and inverted using direct methods. It can, however, be easily shown that if these sub-blocks are replaced by spectrally equivalent approximations, then the rate of convergence of these algorithms remains asymptotically the same. In order to reduce overhead cost, we therefore focus on constructing approximations which are inexpensive to construct,

* Department of Mathematics, University of California at Los Angeles, Los Angeles, CA. 90024.
This work was supported in part by the Department of Energy under contract DE-FG03-87ER25037, by the Army Research Office under contract ARO DAAL03-91-G-150, by the National Science Foundation under grant FDP NSF ASC 9003002, and by the Office for Naval Research under contract ONR N00014-90-J-1695.

and which are inexpensive to invert.

Two kinds of approximations will be considered, one based on Fourier approximations of the interface operators, and another based on sparse algebraic approximation of the interface operators by a *probing* technique. The Fourier based approximations can be shown to be spectrally equivalent with respect to mesh size variations. However, their performance can be sensitive to the coefficients. On the other hand, the probing based algorithms adapt well to the coefficients, but can be sensitive to mesh size variations.

In § 2, we describe the elliptic problem and the Schur complement system on the interface. In § 3, we describe the original versions of the BPS and VS preconditioners for the Schur complement on the interface. In § 4, we describe the two variants, one based on Fourier approximations, and the other based on the *probing* technique. In § 5, we present numerical results comparing the rates of convergence of the various preconditioners.

2. An elliptic problem and its many subdomain decomposition. Here we describe the block structure obtained when a self-adjoint elliptic problem is discretized on a domain Ω partitioned into many non-overlapping subdomains Ω_i with an interface B separating the subdomains. A reduced Schur complement system is derived for the unknowns on the interface. Some properties of this Schur complement system and an iterative procedure for solving the elliptic problem are described.

2.1. Block partition of elliptic problem. We consider the following 2nd order self adjoint elliptic problem on a polygonal domain $\Omega \in R^2$:

$$\left\{ \begin{array}{rcll} -\nabla \cdot (a(x,y)\nabla u) & = & f & \text{in } \Omega \\ u & = & 0 & \text{on } \partial\Omega, \end{array} \right. \tag{1}$$

where $a(x,y) \in R^{2\times 2}$ is a symmetric, uniformly positive definite matrix function having $L^\infty(\Omega)$ entries, and $f \in L^2(\Omega)$.

We assume that the domain Ω is partitioned into N non-overlapping subdomains $\Omega, \cdots, \Omega_N$ of diameter H, which form the elements of a *quasi-uniform* coarse grid triangulation τ^H, see Fig. 1. We also assume that the subdomains Ω_i are refined to produce a fine grid *quasi-uniform* triangulation τ^h having elements of diameter h. Corresponding to the coarse grid and fine grid triangulations, we discretize (1) either by using finite elements, see [14], or by using finite difference methods, see [29], resulting in a symmetric positive definite linear system

$$A_h u_h = f_h, \tag{2}$$

on the fine grid and

$$A_H u_H = f_H, \tag{3}$$

on the coarse grid.

Let I denote the union of the interiors of the subdomains, and let B denote the interface separating the subdomains:

$$I = \cup_i \Omega_i, \quad B \equiv (\cup_i \partial\Omega_i) - \partial\Omega.$$

Then, grouping the unknowns in the interior of the subdomains in the vector u_I and the unknowns on the interface B in the vector u_B, we obtain a reordering of the fine grid problem:

$$\begin{bmatrix} A_{II} & A_{IB} \\ A_{IB}^T & A_{BB} \end{bmatrix} \begin{bmatrix} u_I \\ u_B \end{bmatrix} = \begin{bmatrix} f_I \\ f_B \end{bmatrix}. \tag{4}$$

Here A_{II} corresponds to the coupling between nodes in the interior of the subdomains. For most discretizations, including five point discretizations, the interior nodes in Ω_i are coupled only to the nodes on the interface B, and not to adjacent subdomains. In such cases, $A_{II} \equiv \mathit{blockdiag}(A_{11}, \cdots, A_{NN})$ is a block diagonal matrix.

Eliminating interior unknowns u_I, we obtain u_I in terms of u_B:

$$u_I = A_{II}^{-1}\left(f_I - A_{IB}u_B\right), \tag{5}$$

and substituting this in the 2nd block row of (4) yields an equation for u_B:

$$Su_B = f_B - A_{IB}^T A_{II}^{-1} f_I, \tag{6}$$

where $S = A_{BB} - A_{IB}^T A_{II}^{-1} A_{IB}$ is referred to as the Schur complement or interface matrix. Some properties of the Schur complement will be discussed in § 2.3. First, we will outline a procedure for solving (4).

2.2. Iterative solution of the block partitioned system. System (4) can be solved as follows. First problem (6) is solved for u_B and then (5) is solved for u_I. If direct methods are used to solve (6) then S needs to be computed explicitly, and this can be expensive in general (though this is standard practise in the substructuring methods used to solve linear elasticity problems), since it involves computing the action of A_{II}^{-1} on the all columns of A_{IB}. This can be implemented more efficiently through subassembly, see [27], requiring only as many solves on each Ω_i as there are unknowns on $\partial\Omega_i \cap B$. Even if the matrix S has been assembled, it is often preferable to solve (6) by an iterative method, since direct methods to solve (6) require significant memory storage and computational complexity.

Due to the expense of computing S and solving (6) by direct methods, we consider solving (6) by a preconditioned iterative method such as the conjugate gradient method, see [22], without the explicit construction of S. In this case only matrix vector products with S are required, and each such matrix vector product requires the solution of one problem on each subdomain Ω_i. The Schur complement, however, is ill-conditioned with $\kappa(S) \approx O(h^{-1})$, see [2, 3], and therefore requires a preconditioner M; the construction of efficient preconditioners M for S will be the main focus of this paper.

First, we note that the procedure to solve the linear system (4) by solving the reduced Schur complement system (6) corresponds to a block LU factorization based solution:

$$A = LU = \begin{bmatrix} A_{II} & 0 \\ A_{IB}^T & I \end{bmatrix} \begin{bmatrix} I & A_{II}^{-1}A_{IB} \\ 0 & S \end{bmatrix}, \tag{7}$$

for $S = A_{BB} - A_{IB}^T A_{II}^{-1} A_{IB}$. Thus

$$A^{-1} = \begin{bmatrix} I & -A_{II}^{-1}A_{IB}S^{-1} \\ 0 & S^{-1} \end{bmatrix} \begin{bmatrix} A_{II}^{-1} & 0 \\ -A_{IB}^T A_{II}^{-1} & I \end{bmatrix},$$

and backsolving requires solving two systems with coefficient matrices A_{II} and one system with coefficient matrix S, which will be done using a preconditioned conjugate gradient method. We note that, it is possible to construct a global preconditioner $\tilde{A}$ for A by replacing A_{II} by preconditioner $\tilde{A}_{II}$, and by replacing S by preconditioner M. In this case the inverse of the global preconditioner $\tilde{A}$ has the form:

$$\tilde{A}^{-1} = \begin{bmatrix} I & -\tilde{A}_{II}^{-1} A_{IB} M^{-1} \\ 0 & M^{-1} \end{bmatrix} \begin{bmatrix} \tilde{A}_{II}^{-1} & 0 \\ -A_{IB}^T \tilde{A}_{II}^{-1} & I \end{bmatrix}.$$

Approximations to the submatrices $\tilde{A}_{ii}$ can be obtained for instance by replacing it either with a scaled version of the Laplacian, or by other preconditioners, such as ILU, see [9].

2.3. Some properties of the Schur complement S. The Schur complement matrix S is a discrete approximation to a Steklov-Poincare operator, see [1], which enforces *transmission boundary* conditions on the interface B. In the continuous problem, these transmission boundary conditions correspond to the requirement that the solution u be continuos across the interface and that the flux $\vec{n} \cdot (a(x,y)\nabla u)$ also be continuos across the interface. In the discrete case, the action of the Schur complement on a grid function u_B on B is the same as the action of the discrete operator A_h on the *discrete harmonic extension* of u_B into the subdomains; More specifically, let $E^h u_B$ denote the *discrete harmonic extension* on B to the interior of the subdomains:

$$E^h u_B \equiv \left[-A_{II}^{-1} A_{IB} u_B, u_B \right], \tag{8}$$

then

$$\begin{bmatrix} A_{II} & A_{IB} \\ A_{IB}^T & A_{BB} \end{bmatrix} \begin{bmatrix} -A_{II}^{-1} A_{IB} u_B \\ u_B \end{bmatrix} = \begin{bmatrix} 0 \\ S u_B \end{bmatrix}.$$

Thus, if R_B denotes the pointwise restriction of nodal values of a grid function onto the nodes on B, then $S u_B = R_B A_h E^h u_B$. In addition,

$$x_B^T S x_B = (E x_B)^T A_h (E x_B). \tag{9}$$

This property shows the positive definiteness of the Schur complement. In addition to S being positive definite, it is an M-matrix when A_h is an M-matrix, i.e., $S_{ij} \leq 0$ for $i \neq j$ and $(S^{-1})_{ij} \geq 0$ for all i, j, see [29, 12].

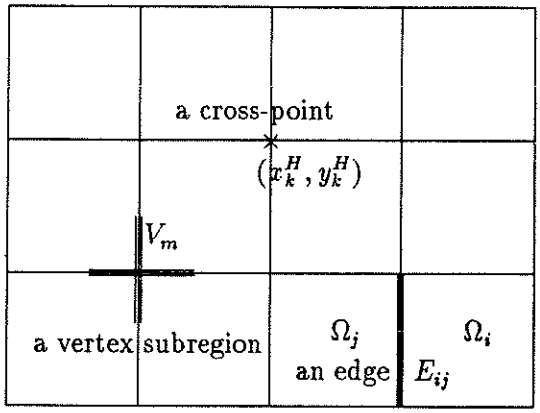
Remark. For finite element discretizations, let $A^{(i)}$ denote the stiffness matrix obtained by integrating the bilinear form on Ω_i, i.e., the discretization of the Neumann problem on Ω_i. For finite difference methods, let $A^{(i)}$ correspond to the discretization with discontinuous coefficients which is $a(x,y)$ in Ω_i and zero outside Ω_i. Then, the energy $x^T A x$ can be partitioned as

$$x^T A_h x = \sum_{i=1}^{N} x^T A^{(i)} x, \tag{10}$$

and correspondingly, the Schur complement S can be partitioned:

$$x_B^T S x_B = \sum_{i=1}^{N} x_B^T S^{(i)} x_B, \tag{11}$$

FIG. 1. *The vertex space partitioning of the interface.*



where

$$S^{(i)} = R_B A^{(i)} E^h. \tag{12}$$

Each $S^{(i)}$ is a map of the Dirichlet values u_B to the *normal derivatives* on $\partial\Omega_i \cap B$ of the discrete harmonic extension $E^h u_B$, and this is not a local operator, i.e., the matrix $S^{(i)}$ is dense on $\partial\Omega_i \cap B$, see [2]. In the two subdomain case, $S = S^{(1)} + S^{(2)}$ is thus a map of the Dirichlet value u_B to the jump in the normal derivatives on B of the discrete harmonic extension $E^h u_B$, which corresponds to the a discrete approximations of the transmission boundary condition. In the two dimensional case, the entries of S decay as $|S_{ij}| = 0(\frac{1}{|i-j|^2})$, see Golub-Mayer [21], and preconditioners for S have been studied extensively, see [2, 8, 4, 19, 10]. The important properties of the two subdomain Schur complement is that its entries decay away from its main diagonal, and that it is uniformly spectrally equivalent to the square root of the Laplace operator on B, as the mesh size goes to zero. Due to this connection, it can be shown that its condition number grows as $\kappa(S) = O(\frac{1}{h})$, see [2]. Applications of both these properties will be discussed in § 4.1 and § 4.2.

3. The BPS and VS preconditioners for S. We will describe two preconditioners for S in this Section, one introduced by Bramble, Pasciak and Schatz (BPS) [3], and another, the vertex space preconditioner (VS) introduced by Smith [27] and Nepomnyaschikh [25]. Both these can be interpreted as generalized block Jacobi type preconditioners for (6) with overlapping blocks and involving residual correction on a coarse grid. Variants of these preconditioners will be discussed in § 4.

3.1. Notations for a partition of the interface B. In the case of many subdomains, the interface B can be partitioned as a union of edges E_{ij} and cross-points V, see Fig. 1:

$$B = \cup_{ij} E_{ij} \quad \cup V,$$

where E_{ij} denotes the edge separating subdomains Ω_i and Ω_j, and V denotes the collection of cross-points (vertices (x_k^H, y_k^H) of the subdomains). Note that the edges E_{ij} are assumed not to include its endpoints.

For each edge E_{ij} we define $R_{E_{ij}}$ as the pointwise restriction of nodal values to E_{ij}, i.e., if g_B is a grid function defined on B, and if E_{ij} contains n_{ij} interior nodes, then its restriction $R_{E_{ij}} g_B$ is a vector with n_{ij} components defined on E_{ij} by

$$R_{E_{ij}} g_B = g_B \text{ on } E_{ij}.$$

Its transpose $R_{E_{ij}}^T$ extends grid functions in E_{ij} by zero to the rest of B:

$$R_{E_{ij}}^T g_{E_{ij}} = \begin{cases} g_{E_{ij}} & \text{on } E_{ij} \\ 0 & \text{on } B - E_{ij} \end{cases}.$$

Similarly, we define R_V as the pointwise restriction map onto the cross-points; if g_B is a grid function on B, and if there are n_V cross-points on B, then $R_V g_B$ is a vector with n_V components defined by

$$R_V g_B = g_B \text{ on } V.$$

Its transpose R_V^T, is thus extension by zero of nodal values in V to B:

$$R_V^T g_V = \begin{cases} g_V & \text{on } V \\ 0 & \text{on } B - V \end{cases}.$$

3.2. The BPS preconditioner. In order to motivate the construction of the BPS preconditioner, we first define a block Jacobi preconditioner M_J consisting of diagonal blocks of the Schur complement S in the following block partitioning of the interface B. Let us suppose there are n edges E_{ij} with some ordering $E_1, \cdots, E_n$. If the unknowns on each edge E_i is grouped together in u_{E_i}, and if the unknowns on the cross-points are grouped in u_V, then S has the following block partitioning corresponding to $(u_{E_1}, \cdots, u_{E_n}, u_V)$:

$$S = \begin{bmatrix} S_{E_1} & \cdots & S_{E_1 E_n} & S_{E_1 V} \\ \vdots & \cdots & \vdots & \vdots \\ S_{E_1 E_n}^T & \cdots & S_{E_n} & S_{E_n V} \\ S_{E_1 V}^T & \cdots & S_{E_n V}^T & S_V \end{bmatrix}.$$

Here, $S_{E_i E_j} \equiv R_{E_i} S R_{E_j}^T$ denotes the coupling in S between nodes on E_i and E_j, and $S_{E_i V} \equiv R_{E_i} S R_V^T$ denotes the coupling in S between nodes on E_i and V. Note that edges E_i and E_j will be coupled in S only if they are part of the boundary of a common subdomain Ω_k. This can be seen by using the relation between Schur complement and discrete harmonic extensions; since, for instance, discrete harmonic extensions of grid functions on edge E_i is non-zero only in the subdomains that for which E_i is part of its boundary. S is thus a block sparse matrix and corresponding to each edge E_{ij}, the submatrix $S_{E_{ij}}$ is identical to the two subdomain Schur complement on interface E_{ij} separating Ω_i and Ω_j. The submatrix S_V which corresponds to coupling in S between cross-points is almost a diagonal matrix since the cross-points are weakly coupled in S. In the case of five point discretizations on rectangular subdomains, S_V is diagonal since the corner nodes (cross-points) of rectangular domains do not influence the solution in the interior.

For this block partition of S, we define the action of the inverse of the block Jacobi preconditioner M_J:

$$M_J^{-1} g_B = \sum_{\text{edges } ij} R_{E_{ij}}^T S_{E_{ij}}^{-1} R_{E_{ij}} f_B + R_V^T S_V^{-1} R_V f_B. \tag{13}$$

This block Jacobi preconditioned system can be shown to have a a condition number satisfying:

$$c_1 H^{-2} \leq \frac{\lambda_{max}(M_J^{-1} S)}{\lambda_{min}(M_J^{-1} S)} \leq c_2 H^{-2}(1 + \log^2(H/h)),$$

where c_1 and c_2 are independent of H and h, see [3, 30]. This indicates that as $H \to 0$, i.e., as the number of subdomains increases, the rate of convergence deteriorates. This can be attributed to the absence of global communication of information amongst all the edges in the preconditioning step.

The original version of the BPS algorithm [3] involves two changes to this block Jacobi preconditioner. One is that the submatrices $S_{E_{ij}}$ are replaced by Fourier based approximations $\tilde{S}_{E_{ij}}$ which will be described in § 4. The second change is to incorporate global coupling in order to obtain a rate of convergence which does not deteriorate as the number of subdomains is increased. In order to do this, the cross-points correction term $R_V^T S_V^{-1} R_V$ in (13) is replaced by a coarse grid correction term $R_H^T A_H^{-1} R_H$ as in two level multigrid methods (involving weighted restriction and interpolation maps R_H and R_H^T respectively). These are defined below. Let $\phi_{k,H}$ denote the kth coarse grid piecewise linear finite element basis function

$$\phi_{k,H}(x_l^H, y_l^H) = \begin{cases} 1 & \text{if } l = k \\ 0 & \text{if } l \neq k \end{cases},$$

where (x_l^H, y_l^H) is the lth cross-point. Then,

$$(R_H f_B)(x_k^H, y_k^H) \equiv \sum_{(x_j, y_j)} \phi_{k,H}(x_j^H, y_j^H) f_B(x_j^H, y_j^H).$$

Its transpose R_H^T thus denotes linear interpolation of the nodal values on the endpoints of edges E_{ij}:

$$\left(R_H^T g_V\right)(x, y) \equiv \sum_k g_V(x_k^H, y_k^H) \phi_{k,H}(x, y), \quad (x, y) \in B.$$

With these changes, the BPS preconditioner can be defined:

$$M_{BPS}^{-1} f_B = \sum_{\text{edges } i,j} R_{E_{ij}}^T \tilde{S}_{E_{ij}}^{-1} R_{E_{ij}} f_B + R_H^T A_H^{-1} R_H f_B.$$

These changes improve the condition number over that of the block Jacobi version.

THEOREM 3.1. *The BPS preconditioner satisfies*

$$\frac{\lambda_{max}(M_{BPS}^{-1} S)}{\lambda_{min}(M_{BPS}^{-1} S)} \leq c_2(1 + \log^2(H/h)),$$

where c_2 is independent of H and h.

Proof. See [3] and [30]. □

Remark. It can be easily verified that for five point discretizations of the Laplacian, the coarse grid Schur complement matrix $S_H \equiv R_H S_h R_H^T$ is equal to the coarse grid discretization $A_H = R_H^T A_h R_H$, since piecewise linear interpolation results in grid functions which are discrete harmonic on the subdomains. In case of more general coefficients, it can be shown that A_H and S_H are spectrally equivalent with respect to coarse grid size H.

3.3. The vertex space algorithm of Smith and Nepomnyaschikh. The logarithmic growth in the condition number of the BPS preconditioner can be attributed to the neglect of coupling between adjacent edges of B. The VS preconditioner of Smith [27] and Nepomnyaschikh [25] incorporates some coupling between adjacent edges through the use of certain overlapping blocks of S corresponding to nodes on certain *vertex regions* V_k, which will be defined, and it leads to a condition number independent of mesh parameters.

Let V_k denote the portion of B within a distance of βH from (x_k^H, y_k^H) for some positive fraction $0 < \beta < 1$, see Fig. 1. We refer to each V_k as a vertex region or vertex space. We define the corresponding pointwise restriction map R_{V_k} to be

$$R_{V_k} g_B = g_B \text{ on } V_k.$$

Its transpose $R_{V_k}^T$ is thus extension by zero outside V_k:

$$R_{V_k}^T g_{V_k} = \begin{cases} g_{V_k} & \text{on } V_k \\ 0 & \text{on } B - V_k. \end{cases}$$

Corresponding to each vertex region V_k, the submatrix S_{V_k} is defined by $S_{V_k} = R_{V_k} S R_{V_k}^T$. The action of the inverse of the vertex space preconditioner M_{VS} involves the inversion of these new overlapping blocks in addition to the blocks used in the BPS preconditioner:

$$(14) \quad M_{VS}^{-1} f_B = R_H^T A_H^{-1} R_H f_B + \sum_{E_{ij}} R_{E_{ij}}^T (S_{E_{ij}})^{-1} R_{E_{ij}} f_B + \sum_{V_k} R_{V_k}^T (S_{V_k})^{-1} R_{V_k} f_B.$$

The following result is proved in [27, 25].

THEOREM 3.2. *Suppose the overlap of the vertex regions V_k is βH, then:*

$$\frac{\lambda_{max}(M_{VS}^{-1} S)}{\lambda_{min}(M_{VS}^{-1} S)} \leq C(\beta),$$

where $C(\beta)$ is independent of H and h.

Remark. Other bounds are available for the condition number of the vertex space preconditioned system:

$$\frac{\lambda_{max}(M_{VS}^{-1} S)}{\lambda_{min}(M_{VS}^{-1} S)} \leq \begin{cases} c_1(1 + c_2(1/\beta)), \\ c_3(1 + \log^2(H/h)), \end{cases}$$

where c_1, c_2 are independent of H and h, but may possibly depend on the coefficients $a(x, y)$, while c_3 is independent of H, h and the coefficients $a(x, y)$ provided the coefficients are constant in each subdomain Ω_i, see [27, 31, 17].

4. Two variants of the vertex space method. An important consideration in the implementation of the algorithms is the expense of computing the edge and vertex matrices $S_{E_{ij}}$ and S_{V_k}, respectively, and the cost of solving the subproblems using direct methods. If there are n_i nodes on each $\partial\Omega_i \cap B$, then computing all the submatrices $S_{E_{ij}}$ and S_{V_k} would require solving n_i problems on each Ω_i, and this increases as the mesh size h is reduced. If n_{ij} is the number of nodes on E_{ij}, the cost of using direct methods to solve edge problems is $O(n_{ij}^2)$ once the Cholesky factorizations have been determined, see [27, 28], since the edge submatrices $S_{E_{ij}}$ are dense. n_{ij} increases as the mesh size h is reduced.

This expense can be significantly reduced if the exact edge and vertex matrices are replaced by approximations which can be computed at significantly less cost, and which can be inverted at less cost. If these approximations are spectrally equivalent to the exact submatrices, then the overall preconditioner would remain spectrally equivalent to the exact VS preconditioner, and the number of iterations required to solve (6) would remain independent of h, see Theorem 4.3.

In this Section, we describe two variants of the vertex space and BPS algorithms in which the exact edge and vertex matrices are replaced by approximations. One variant is based on Fourier approximations of both the edge and vertex matrices, while the other variant is based on sparse algebraic approximation of both these matrices using a *probing* technique. Combinations of Fourier and probe approximations are also possible, but will not be considered here for simplicity, e.g., see [11].

4.1. Fourier approximations. Fourier based approximations of the edge and vertex matrices are constructed based on the property that, restricted to simple curves (curves which do not intersect themselves), the Schur complement is spectrally equivalent to the square root of the Laplace operator on it, and this has been studied extensively, see [2, 21, 8, 4, 19, 10].

4.1.1. Fourier edge approximations. First, we consider Fourier approximations of the edge matrices $S_{E_{ij}}$. Let edge E_{ij} separate Ω_i and Ω_j. Since, the submatrix $S_{E_{ij}}$ is identical to the two subdomain Schur complement on E_{ij}, standard preconditioners for the two subdomain case can be applied, see [2, 21, 8, 4, 19, 10].

Let J denote the discrete Laplacian on a uniform grid containing n_{ij} interior nodes with mesh size $h = 1/(n_{ij}+1)$:

$$-h^2\frac{d^2}{dx^2} \approx J \equiv \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}.$$

Then, $J^{1/2}$ is uniformly spectrally equivalent to $S_{E_{ij}}$ as the mesh size h is varied, see [2]. Since the discrete Laplacian is diagonalized by the sine transform, $J = W\Lambda W$, where

$$W_{ij} = \sqrt{2h}\sin(ij\pi h),$$

and $\Lambda = diag(\lambda_i)$ with $\lambda_i = 4\sin^2(\frac{i\pi h}{2})$, it follows that $J^{1/2} = W\Lambda^{1/2}W$. By using Fast Sine Transforms, it is possible to compute the action of the inverse of $J^{1/2}$ in $O(n_{ij}\log(n_{ij}))$ flops.

The Fourier based preconditioners M considered here are all based on the sine transform W, but vary with the choice of eigenvalues:

$$M = W\,diag(\mu_k)W.$$

The eigenvalues μ_k are chosen to better approximate the eigenvalues of the exact Schur complement $S_{E_{ij}}$. In the model case of Laplaces equation on $\Omega_i \cup \Omega_j$ with rectangular subdomains $\Omega_i = [0,1]\times[0,l_i]$ and $\Omega_j = [0,1]\times[-l_j,0]$, where m_i and m_j are positive integers with $l_i = (m_i+1)h$ and $l_j = (m_j+1)h$, the eigendecomposition of the Schur complement is known exactly. These exact eigenvalues are given below in M_{Chan}, along with the eigenvalues of three other preconditioners:

Dryja preconditioner M_D, see [16]:	$\mu_k = \lambda_k^{1/2}$
Golub-Mayers preconditioner M_{GM}, see [21]:	$\mu_k = \sqrt{\lambda_k + \frac{1}{4}\lambda_k^2}$
BPS preconditioner M_{BPS}, see [3]:	$\mu_k = \sqrt{\lambda_k(1-\frac{\lambda_k}{6})}$
Chan preconditioner M_{Chan}, see [8]:	$\mu_k = \left(\frac{1-\gamma_k^{m_i+1}}{1+\gamma_k^{m_i+1}} + \frac{1-\gamma_k^{m_j+1}}{1+\gamma_k^{m_j+1}}\right)\sqrt{\lambda_k + \frac{1}{4}\lambda_k^2}$ where $\gamma_k = \frac{1+\frac{\lambda_k}{2}-\sqrt{\lambda_k+\frac{\lambda_k^2}{4}}}{1+\frac{\lambda_k}{2}+\sqrt{\lambda_k+\frac{\lambda_k^2}{4}}}$.

(15)

We have the following result.

LEMMA 4.1. *Let M denote either the Dryja, Golub-Mayer, BPS or Chan preconditioners for $S_{E_{ij}}$. Then*

$$\frac{\lambda_{max}(M^{-1}S_{E_{ij}})}{\lambda_{min}(M^{-1}S_{E_{ij}})} \le C_1,$$

where C_1 is independent of h. For $S_{E_{ij}}$ corresponding to Laplaces equation on the model domain $\Omega_i \cup \Omega_j$ with rectangular subdomains $\Omega_i = [0,1]\times[0,l_i]$ and $\Omega_j = [0,1]\times[-l_j,0]$, the condition number of the Dryja, Golub-Mayer and BPS preconditioners satisfy:

$$\frac{\lambda_{max}(M^{-1}S_{E_{ij}})}{\lambda_{min}(M^{-1}S_{E_{ij}})} \le C_2(1+\frac{1}{l_i}+\frac{1}{l_j}),$$

where C_2 is independent of h, l_i and l_j, while the condition number of the Chan preconditioner satisfies:

$$\frac{\lambda_{max}(M_{Chan}^{-1}S_{E_{ij}})}{\lambda_{min}(M_{Chan}^{-1}S_{E_{ij}})} \le C_2.$$

Proof. See Bjorstad and Widlund [2], Chan [8]. □

The Fourier preconditioners described so far do not depend on the coefficients $a(x,y)$ of the elliptic problem, and thus the rate of convergence can be sensitive to the coefficients, see [12]. In order to incorporate some information about the coefficients, we scale the Fourier preconditioners by a scaling matrix. In the original BPS algorithm [3], a scalar coefficient α_{ij} representing the average of the eigenvalues of $a(x,y)$ at a point in Ω_i and a point in Ω_j was used as scaling on each edge E_{ij}. Here, we use a diagonal matrix D_{ij} as scaling, where D_{ij} denotes the diagonal of A_h restricted to E_{ij}, and define the diagonally scaled Fourier preconditioners by

$$\tilde{S}_{E_{ij}}^F \equiv D_{ij}^{1/2} W\,diag(\mu_k) W\,D_{ij}^{1/2}. \tag{16}$$

For most applications to isotropic coefficients, these diagonally scaled Fourier preconditioners perform well.

4.1.2. Fourier vertex space approximation. Next, we describe approximations of the vertex space matrices S_{V_k} based on Fourier techniques. For the case of the discrete Laplacian, it is possible to express the eigendecomposition of S_{V_k} for cross shaped vertex regions in terms of sine transforms, thereby enabling the use of fast transforms to invert S_{V_k}, see [25]. However, it is not easily generalized to the case of varying coefficients, and instead we construct approximations to the vertex matrices by using a direct sum of smaller matrices that will be described in the following.

We will describe the procedure for the model geometry of Fig. 3. Let u_{V_k} be a grid function on B which is zero outside the vertex region V_k, i.e., zero on $B - V_k$. Then, by the property of the Schur complement (11), we obtain that

$$u_{V_k}^T S_{V_k} u_{V_k} = \sum_{i=1}^{4} u_{V_k}^T S^{(i)} u_{V_k}, \tag{17}$$

where $S^{(i)}$ is the component of the Schur complement originating from Ω_i, as described in (12). For $i = 1, 2, 3, 4$, let L_i^k denote the L-shaped segment $V_k \cap \partial\Omega_i$, and further let $R_{L_i^k}$ denote the pointwise restriction onto L_i^k. Then, as in the case for the edges, $(R_{L_i^k} u_B)^T S^{(i)} (R_{L_i^k} u_B)$, is spectrally equivalent to $(R_{L_i^k} u_B)^T M_i^k (R_{L_i^k} u_B)$ where M_i^k is any of the unscaled Fourier approximations to the square root of the Laplacian on L_i^k, see (15). Let D_i^k denote the diagonal of $A^{(i)}$ restricted to L_i^k. Then, by including the effects of coefficients, we define the following scaled Fourier based preconditioner for S_{V_k}:

$$\tilde{S}_{V_k}^F \equiv \sum_{i=1}^{4} R_{L_i^k}^T (D_i^k)^{1/2} M_i^k (D_i^k)^{1/2} R_{L_i^k}. \tag{18}$$

For most applications we considered, it was sufficient to choose the number of nodes on the vertex regions V_k to be small, say 5 or 9, and so the matrices $\tilde{S}_{V_k}^F$ can be computed at little expense, and can be inverted inexpensively by direct methods.

THEOREM 4.2. *The matrices $\tilde{S}_{V_k}^F$ are spectrally equivalent to to S_{V_k}, i.e., there exists constant c_0, c_1 independent of h such that*

$$c_0 \leq \frac{\lambda_{max}\left((\tilde{S}_{V_k}^F)^{-1} S_{V_k}\right)}{\lambda_{min}\left((\tilde{S}_{V_k}^F)^{-1} S_{V_k}\right)} \leq c_1.$$

Proof. The proof follows trivially by application of the standard result, see [2, 4], that on a simple edge such as L_i^k, the square root of the Laplacian on it is spectrally equivalent to the energy of the local Schur complement, i.e., there exists constants $c_0^{(i)}, c_1^{(i)}$ independent of h such that:

$$c_0^{(i)} \leq \frac{x_{V_k}^T S_{V_k}^{(i)} x_{V_k}}{x_{V_k}^T M_i^k x_{V_k}} \leq c_1^{(i)}.$$

Similar bounds hold when M_i^k is replaced by $(D_i^k)^{1/2} M_i^k (D_i^k)^{1/2}$, with suitably modified constants $c_0^{(i)}, c_1^{(i)}$, since the entries of D_i^k can be bounded in terms of the upper and

lower bounds for $a(x,y)$ in the neighborhood of L_i^k, independent of h. From this the result follows immediately, since:

$$\min_i\{c_0^{(i)}\} \leq \frac{\sum_{i=1}^4 x_{V_k}^T S_{V_k}^{(i)} x_{V_k}}{\sum_{i=1}^4 x_{V_k}^T M_i^k x_{V_k}} \leq \max_i\{c_1^{(i)}\},$$

for the suitably modified coefficients $c_0^{(i)}$ and $c_1^{(i)}$. □

4.1.3. Fourier based preconditioner. Based on the approximations $\tilde{S}_{E_{ij}}^F$ and $\tilde{S}_{V_k}^F$, we define the Fourier vertex space preconditioner (FVS) by

$$M_{FVS}^{-1} \equiv R_H^T A_H^{-1} R_H + \sum_{ij} R_{E_{ij}}^T (\tilde{S}_{E_{ij}}^F)^{-1} R_{E_{ij}} + \sum_k R_{V_k}^T (\tilde{S}_{V_k}^F)^{-1} R_{V_k}, \tag{19}$$

and the Fourier BPS preconditioner (FBPS) by, see [3]:

$$M_{FBPS}^{-1} \equiv R_H^T A_H^{-1} R_H + \sum_{ij} R_{E_{ij}}^T (\tilde{S}_{E_{ij}}^F)^{-1} R_{E_{ij}}. \tag{20}$$

Note that the Fourier edge approximations $\tilde{S}_{E_{ij}}^F$ can be inverted in $O(n_{ij}\log(n_{ij}))$ flops, using the Fast Sine Transform. Direct methods can be used to solve the Fourier vertex problems $\tilde{S}_{V_k}^F$. The coarse grid matrix problem A_H can be solved using either direct or iterative methods.

Remark. In the original BPS preconditioner [3], the edge approximations were chosen to be

$$\tilde{S}_{E_{ij}} = \alpha_{ij} W diag(\mu_k) W,$$

where α_{ij} is the average of the eigenvalues of $a(x,y)$ at a point from Ω_i and a point from Ω_j, and $\mu_k = \sqrt{\lambda_k(1-\lambda_k/6)}$, and this differs from the version described in (20) because of the scaling matrix D_i^k.

THEOREM 4.3. *The Fourier preconditioner M_{FVS} satisfies:*

$$c_0 \leq \frac{\lambda_{max}(M_{FVS}^{-1} S)}{\lambda_{min}(M_{FVS}^{-1} S)} \leq c_1,$$

where c_0, c_1 are independent of H, h, but may depend on the overlap ratio β.

Proof. Bounds for the extreme eigenvalues of $M_{FVS}^{-1} S$ is obtained from bounds for the Rayleigh quotient:

$$\lambda_{min}(M_{FVS}^{-1} S) \leq \left(\frac{x_B^T S x_B}{x_B^T M_{VS} x_B}\right)\left(\frac{x_B^T M_{VS} x_B}{x_B^T M_{FVS} x_B}\right) \leq \lambda_{max}(M_{FVS}^{-1} S).$$

The fraction $x_B^T S x_B / x_B^T M_{VS} x_B$ has uniform upper and lower bounds, see [27]. It therefore suffices to obtain uniform upper and lower bounds for the fraction $x_B^T M_{VS} x_B / x_B^T M_{FVS} x_B$ or equivalently for

$$\lambda_{min}(M_{FVS} M_{VS}^{-1}) \leq \frac{x_B^T M_{VS}^{-1} x_B}{x_B^T M_{FVS}^{-1} x_B} \leq \lambda_{max}(M_{FVS} M_{VS}^{-1}).$$

By spectral equivalence of the edge Fourier approximations, Lemma 4.1, there exists constants c_{ij} and C_{ij} independent of H and h such that:

$$c_{ij} \leq \frac{x_B^T S_{E_{ij}}^{-1} x_B}{x_B^T (\tilde{S}_{E_{ij}}^F)^{-1} x_B} \leq C_{ij},$$

and similarly for the vertex spaces, by Theorem 4.2, there exists constants c_k and C_k independent of H and h such that:

$$c_k \leq \frac{x_B^T S_{V_k}^{-1} x_B}{x_B^T (\tilde{S}_{V_k}^F)^{-1} x_B} \leq C_k.$$

Letting $C = \max\{C_{ij}, C_k\}$ and $c = \min\{c_{ij}, c_k\}$, we obtain that

$$c \leq \frac{\sum_{ij} x_B^T S_{E_{ij}}^{-1} x_B + \sum_k x_B^T S_{V_k}^{-1} x_B + x_B^T R_H^T A_H^{-1} R_H x_B}{\sum_{ij} x_B^T (\tilde{S}_{E_{ij}}^F)^{-1} x_B + \sum_k x_B^T (\tilde{S}_{E_k}^F)^{-1} x_B + x_B^T R_H^T A_H^{-1} R_H x_B} \leq C,$$

and hence our result follows. □

4.2. Probe approximations. Next, we describe another variant of the VS and BPS preconditioners in which the edge and vertex matrices are approximated by sparse matrices obtained using an extension of the *probing* technique of Chan and Resasco [13], Keyes and Gropp [23, 24], and Eisenstat [18]. Unlike Fourier based approximations, the construction of the probe approximations require solving six problems on each subdomain, and thus has a greater overhead cost than the Fourier approximations, but still considerably less than the exact submatrices. An advantage of these approximations is that they often adapt well to coefficient variations and aspect ratios. However a disadvantage is that they do not adapt optimally to mesh size variations.

We will describe construction of these probe approximations for the model rectangular geometry of Fig. 1. The techniques are easily extended to more general geometries.

4.2.1. Edge probe approximations. We first describe how sparse approximations to the edge matrices can be constructed [13]. In its basic form, the *probing* technique consists of approximating each $S_{E_{ij}}$ by a tridiagonal matrix $\tilde{S}_{E_{ij}}$ which is chosen on the assumption that each node on an edge is strongly coupled in S only to nodes adjacent to it and weakly coupled to the other nodes. A heuristic motivation for this is that the entries of each $S_{E_{ij}}$ are known to decay rapidly away from the main diagonals:

$$|(S_{E_{ij}})_{lm}| = O\left(\frac{1}{|l-m|^2}\right),$$

see Golub and Mayer [21].

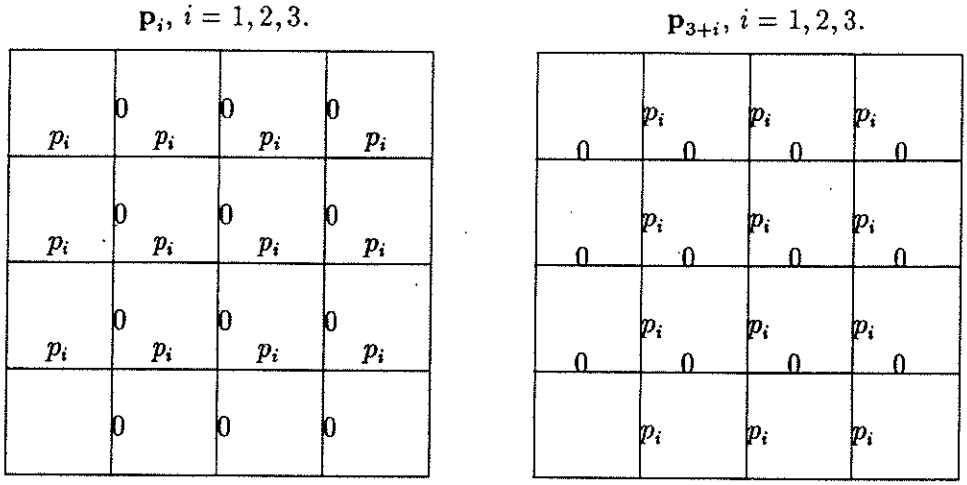
To obtain a tridiagonal approximation $\tilde{S}_{E_{ij}}$ to $S_{E_{ij}}$, we equate the matrix vector products $S_{E_{ij}} p_i$ to $\tilde{S}_{E_{ij}} p_i$ for the following three *probe* vectors p_i:

$$p_1 = [1,0,0,1,0,0,\cdots]^T, \quad p_2 = [0,1,0,0,1,0,\cdots]^T, \quad p_3 = [0,0,1,0,0,1,\cdots]^T.$$

These matrix vector products $[\tilde{S}_{E_{ij}} p_1, \tilde{S}_{E_{ij}} p_2, \tilde{S}_{E_{ij}} p_3]$ results in:

$$\begin{bmatrix} (\tilde{S}_{E_{ij}})_{11} & (\tilde{S}_{E_{ij}})_{12} & & \\ (\tilde{S}_{E_{ij}})_{21} & (\tilde{S}_{E_{ij}})_{22} & (\tilde{S}_{E_{ij}})_{23} & \\ & (\tilde{S}_{E_{ij}})_{32} & (\tilde{S}_{E_{ij}})_{33} & \ddots \\ & & \ddots & \ddots \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ \vdots & \vdots & \vdots \end{bmatrix} = \begin{bmatrix} (\tilde{S}_{E_{ij}})_{11} & (\tilde{S}_{E_{ij}})_{12} & 0 \\ (\tilde{S}_{E_{ij}})_{21} & (\tilde{S}_{E_{ij}})_{22} & (\tilde{S}_{E_{ij}})_{23} \\ (\tilde{S}_{E_{ij}})_{34} & (\tilde{S}_{E_{ij}})_{32} & (\tilde{S}_{E_{ij}})_{33} \\ \vdots & \vdots & \vdots \end{bmatrix},$$

FIG. 2. *Simultaneous probe vectors*



and equating this with $[S_{E_{ij}}p_1, S_{E_{ij}}p_2, S_{E_{ij}}p_3]$ gives:

$$\begin{bmatrix} (\tilde{S}_{E_{ij}})_{11} & (\tilde{S}_{E_{ij}})_{12} & 0 \\ (\tilde{S}_{E_{ij}})_{21} & (\tilde{S}_{E_{ij}})_{22} & (\tilde{S}_{E_{ij}})_{23} \\ (\tilde{S}_{E_{ij}})_{34} & (\tilde{S}_{E_{ij}})_{32} & (\tilde{S}_{E_{ij}})_{33} \\ \vdots & \vdots & \vdots \end{bmatrix} := [S_{E_{ij}}p_1, S_{E_{ij}}p_2, S_{E_{ij}}p_3], \tag{21}$$

from which the non-zero entries of $\tilde{S}_{E_{ij}}$ can be easily read off. In general, $\tilde{S}_{E_{ij}}$ will not preserve the symmetry of $S_{E_{ij}}$, and so we symmetrize it to obtain $\tilde{S}^P_{E_{ij}}$ using a *minimum-modulus* procedure described below:

$$(\tilde{S}^P_{E_{ij}})_{ij} \equiv \begin{cases} (\tilde{S}_{E_{ij}})_{ji} & \text{if } |(\tilde{S}_{E_{ij}})_{ji}| \leq |(\tilde{S}_{E_{ij}})_{ij}| \\ (\tilde{S}_{E_{ij}})_{ij} & \text{if } |(\tilde{S}_{E_{ij}})_{ij}| \leq |(\tilde{S}_{E_{ij}})_{ji}|. \end{cases}$$

We will denote the construction of $\tilde{S}^P_{E_{ij}}$ from $S_{E_{ij}}p_1, S_{E_{ij}}p_2, S_{E_{ij}}p_3$ by the notation:

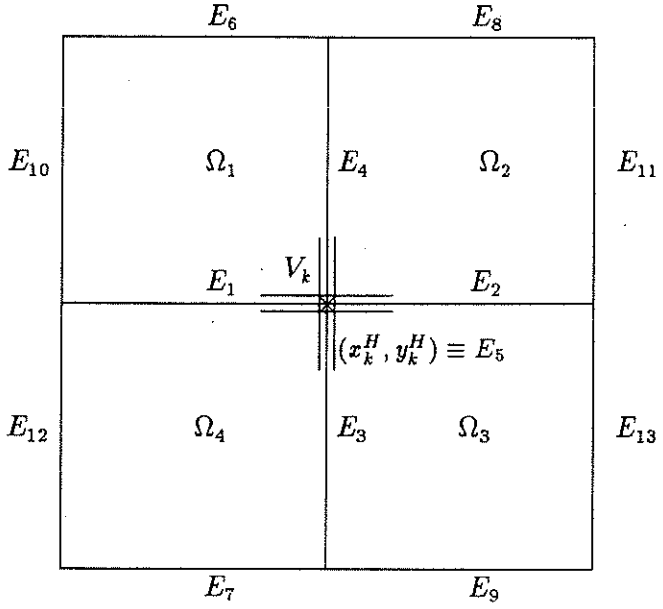
$$\tilde{S}^P_{E_{ij}} = \text{PROBE}(S_{E_{ij}}p_1, S_{E_{ij}}p_2, S_{E_{ij}}p_3). \tag{22}$$

The resulting approximations can be shown to preserve row-wise diagonal dominance, see [12]. This idea is motivated by Curtis, Powell, and Reid [15]. In an analogous way, using a symmetrised variant of [15], see Powell and Toint [26], it is possible to obtain a symmetric tridiagonal approximation directly using just two probe vectors, see [23, 24].

Computing the three matrix vector products $S_{E_{ij}}p_i$ requires three solves on each subdomain Ω_i and Ω_j. Thus, in order to compute edge approximations $\tilde{S}^P_{E_{ij}}$ on the edges of all the subdomains, twelve solves on each subdomain would be required, since the boundary of rectangular subdomains consists of four edges.

We now describe a procedure for computing all the edge approximations using only six solves on each subdomain, by simultaneously prescribing boundary conditions on other edges, an idea first used in Keyes and Gropp [23, 24]. To minimize the

FIG. 3. *Numbering of Edges.*



approximation errors arising from the coupling between vertical and horizontal edges, we will specify probe vectors p_i either on all horizontal edges simultaneously, or on all vertical edges simultaneously. For $i = 1, 2, 3$, see Fig. 2, define:

$$\mathbf{p}_i \equiv \begin{cases} p_i & \text{on all horizontal edges} \\ 0 & \text{on all vertical edges} , \end{cases}$$

$$\mathbf{p}_{3+i} \equiv \begin{cases} 0 & \text{on all horizontal edges} \\ p_i & \text{on all vertical edges.} \end{cases}$$

On the horizontal edges, the probe vectors p_i can be ordered from left to right, and on vertical edges from bottom to top. For these six probe vectors, we compute the discrete harmonic extensions $E^h\mathbf{p}_i = (-A_{II}^{-1}A_{IB}\mathbf{p}_i, \mathbf{p}_i)$, and this involves six solves on each subdomain. If E_{ij} is an horizontal edge, we define:

$$\tilde{S}_{E_{ij}}^P = \text{PROBE}(R_{E_{ij}}A_hE^h\mathbf{p}_1, R_{E_{ij}}A_hE^h\mathbf{p}_2, R_{E_{ij}}A_hE^h\mathbf{p}_3).$$

If E_{ij} is a vertical edge, then we define:

$$\tilde{S}_{E_{ij}}^P = \text{PROBE}(R_{E_{ij}}A_hE^h\mathbf{p}_4, R_{E_{ij}}A_hE^h\mathbf{p}_5, R_{E_{ij}}A_hE^h\mathbf{p}_6).$$

We have the following result on the non-singularity and diagonal dominance of the resulting probe approximations.

THEOREM 4.4. *If the coefficient matrix A_h for the model rectangular geometry of Fig. 1 satisfies the discrete strong maximum principle (as is the case for standard five point discretizations), then the probe approximations $\tilde{S}_{E_{ij}}^P$ obtained above are strictly diagonally dominant.*

Proof. We will prove the diagonal dominance of approximation $\tilde{S}_{E_1}^P$ on edge E_1 in the model geometry of Fig. 3; the proof for the other edge approximations are

analogous. By construction,

$$\tilde{S}^P_{E_1} = \text{PROBE}(R_{E_1} A_h E^h \mathbf{p}_1, R_{E_1} A_h E^h \mathbf{p}_2, R_{E_1} A_h E^h \mathbf{p}_3).$$

Due to the effects of the boundary conditions on the adjacent edges, it is easily verified that (see § 3.2 for notation):

$$R_{E_1} A_h E^h \mathbf{p}_i = S_{E_1} p_i + S_{E_1 E_6} p_i + S_{E_1 E_7} p_i, \text{ for } i = 1, 2, 3,$$

and from this we obtain:

$$\begin{array}{lcl} (\tilde{S}^P_{E_1})_{i,i} & \equiv & \sum_{mod(i-j,3)=0} (S_{E_1} + S_{E_1 E_6} + S_{E_1 E_7})_{i,j}, \\ (\tilde{S}^P_{E_1})_{i,i-1} & \equiv & \sum_{mod(i-j,3)=1} (S_{E_1} + S_{E_1 E_6} + S_{E_1 E_7})_{i,j}, \\ (\tilde{S}^P_{E_1})_{i,i+1} & \equiv & \sum_{mod(i-j,3)=-1} (S_{E_1} + S_{E_1 E_6} + S_{E_1 E_7})_{i,j}. \end{array} \tag{23}$$

For discretizations A_h satisfying the discrete strong maximum principle, S is a diagonally dominant M-matrix, see [12], and so its off diagonal entries are non-positive, and its row sums are non-negative. Using this in (23) we obtain that $(\tilde{S}^P_{E_1})_{ij} \leq 0$ for $j \neq i$ and the row sum:

$$(\tilde{S}^P_{E_1})_{i,i-1} + (\tilde{S}^P_{E_1})_{i,i} + (\tilde{S}^P_{E_1})_{i,i+1} = \sum_j (S_{E_1} + S_{E_1 E_6} + S_{E_1 E_7})_{i,j} > 0,$$

which shows that diagonal dominance is preserved. Finally, the *min-mod* procedure preserves diagonal dominance by definition. □

4.2.2. Probe vertex approximations. Next, we describe how sparse algebraic approximations to the vertex submatrices S_{V_k} can be constructed. Unlike the tridiagonal edge approximations $\tilde{S}^P_{E_{ij}}$ which enabled the use of fast direct solvers, the sparse approximations of the vertex matrices are usually small in general and will be solved by direct methods that do not make use of the sparsity of the matrices. The procedure we will describe results from a slight modification of a technique described in [11]. This new variant can be proved to result in non-singular approximations which preserve diagonal dominance.

For simplicity, we will describe this procedure for the vertex region V_k in the center of the subdomains $\Omega_1, \cdots, \Omega_4$ of Fig. 3. We partition V_k into five disjoint regions:

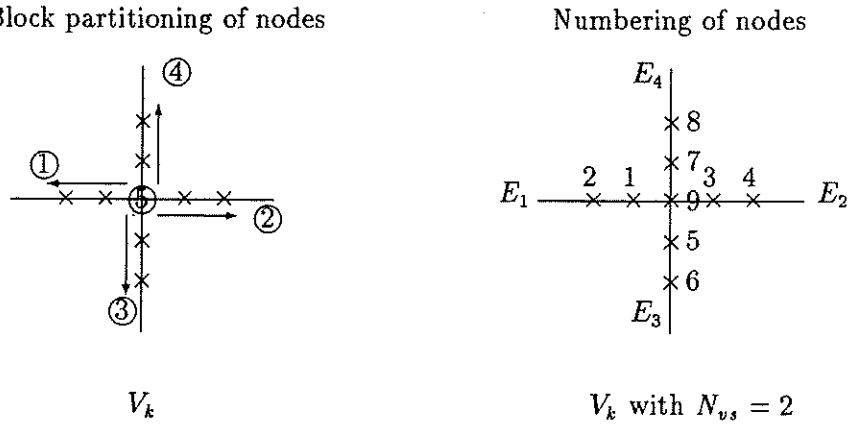
$$V_k = (V_k \cap E_1) \cap (V_k \cap E_2) \cap (V_k \cap E_3) \cap (V_k \cap E_4) \cap (x^H_k, y^H_k), \tag{24}$$

and we obtain a corresponding 5×5 block partition of the vertex matrix S_{V_k}:

$$S_{V_k} = \begin{bmatrix} S_{11} & 0 & S_{13} & S_{14} & S_{15} \\ 0 & S_{22} & S_{23} & S_{24} & S_{25} \\ S^T_{13} & S^T_{23} & S_{33} & 0 & S_{35} \\ S^T_{14} & S^T_{24} & 0 & S_{44} & S_{45} \\ S^T_{15} & S^T_{25} & S^T_{35} & S^T_{45} & S_{55} \end{bmatrix},$$

where each S_{ij} corresponds to the coupling between nodes in block i and block j. The submatrices S_{12} and S_{34} and their transposes are zero, since there is no coupling in S between nodes in E_1 and E_2, and between nodes in E_3 and E_4. We will construct a vertex matrix approximation $\tilde{S}^P_{V_k}$ having the same block structure as S_{V_k}, with subblocks $\tilde{S}_{ij}$ which will be chosen to be sparse.

FIG. 4. *Ordering of unknowns within each vertex subregion V_k*



To facilitate description of the sparsity pattern, we will use the following ordering of nodes within V_k; for each of the four edge segments $E_i \cap V_k$, the nodes will be numbered to increase away from the cross-point (x_k^H, y_k^H), which is ordered last. This ordering is shown in Fig. 4 when each segment $E_i \cap V_k$ contain just two nodes.

Our choice of the sparsity pattern for the sub-blocks $\tilde{S}_{ij}$ is based on the assumption that the elements of S_{V_k} decay with increasing distance between nodes.

Definition and computation of the edge blocks $\tilde{S}_{ii}$ for $i = 1,2,3,4$. Within each edge segment $E_i \cap V_k$ we assume the coupling in S_{V_k} is strong only between adjacent nodes. Based on this assumption, S_{ii} will be approximated by tridiagonal matrices $\tilde{S}_{ii}$ which are chosen to be the submatrices of the tridiagonal edge matrices $\tilde{S}_{E_i}^P$ for $i = 1,2,3,4$, which were computed in § 4.2.

Definition and computation of the blocks $\tilde{S}_{i5}$ for $i = 1,\cdots,5$. We assume the cross-point (x_k^H, y_k^H) is coupled strongly in S_{V_k} only to the nodes adjacent it. Based on this assumption, we choose the vectors $\tilde{S}_{i5}$ to have zero entries except in the first entry:

$$\tilde{S}_{i5} = \begin{bmatrix} (\tilde{S}_{i5})_1 \\ 0 \\ \vdots \end{bmatrix}, \quad \text{for } i = 1,\cdots,5.$$

For five point discretizations on the rectangular geometry of Fig. 1, it can easily be shown that the last row and column of S_{V_k} is exactly equal to the last row and column of $R_{V_k} A_h R_{V_k}^T$, the matrix A_h restricted to V_k. Therefore, we define

$$\begin{aligned} \tilde{S}_{i5} \equiv A_{i5} = S_{i5}, \quad i = 1,\cdots,5 \\ \tilde{S}_{5i} \equiv A_{5i} = S_{5i}, \quad i = 1,\cdots,5. \end{aligned}$$

To see that $A_{i5} = S_{i5}$, first note that S_{i5} is equal to the restriction of Su_B to the ith edge of V_k, where u_B corresponds to boundary data which is 1 on the kth vertex, and zero elsewhere. Now, recall that $Su_B = R_B A_h E^h u_B$. For five point discretizations on rectangular subdomains, the boundary conditions on the corner nodes do not influence

the solution in the interior. Consequently, the discrete harmonic extension $E^h u_B$ is zero in the interior of subdomains, and $A_h E^h u_B$ simply gives the column of A_h corresponding to the kth vertex. Thus $S_{i5} = A_{i5}$.

Definition and computation of $\tilde{S}_{ij}$ for $i = 1, 2$ and $j = 3, 4$. We assume the couplings in S_{V_k} between edge segments $E_i \cap V_k$ and $E_j \cap V_k$ is strong only between the nodes which are closest (adjacent) to the cross-point (x_k^H, y_k^H). Based on this assumption, we choose the submatrices $\tilde{S}_{13}$, $\tilde{S}_{14}$, $\tilde{S}_{23}$ and $\tilde{S}_{24}$ and their transposes to have all zero entries except for the $(1,1)$-th entry.

$$\tilde{S}_{ij} = \begin{bmatrix} (\tilde{S}_{ij})_{11} & 0 & \cdots \\ 0 & 0 & \cdots \\ \cdots & \cdots & \cdots \end{bmatrix}, \quad \text{for } i = 1, 2; \quad j = 3, 4.$$

So there are only eight non-zero entries to define.

Consider for example the entry $(\tilde{S}_{14})_{11}$, which we would like to be an approximation to $(S_{14})_{11}$, the coupling in S between node $(x_k^H - h, y_k^H)$ and node $(x_k^H, y_k^H + h)$. Note that $(S_{14})_{11} = (S\delta_k)(x_k^H - h, y_k^H)$ (i.e. the component of $S\delta_k$ corresponding to the point $(x_k^H - h, y_k^H)$) where δ_k is the boundary data which is 1 on $(x_k^H, y_k^H + h)$ and zero elsewhere, and therefore computing $(S_{14})_{11}$ requires one subdomain solve. In order to reduce this overhead, we would like to extract an approximation from the subdomain solves we already used for the probe edge approximations. For example, one could define $(\tilde{S}_{14})_{11} = (S\mathbf{p}_4)(x_k^H - h, y_k^H)$. However, it turns out that this definition can lead to a non-diagonally dominant (and possibly singular) $\tilde{S}_{V_k}$. This can be seen by noting that

$$(S\mathbf{p}_4)(x_k^H - h, y_k^H) = (S_{E_1E_4}p_1 + S_{E_1E_{10}}p_1 + S_{E_1E_3}p_1 + S_{E_1E_{12}}p_1)(x_k^H - h, y_k^H).$$

The last two terms on the right corresponds to extra influence from Ω_4 on the coupling between nodes $(x_k^H - h, y_k^H)$ and $(x_k^H, y_k^H + h)$ (which should only involve couplings within Ω_1). These extra couplings could cause loss of diagonal dominance, since, in case the coefficients are large in Ω_4, the last two terms will dominate the sum on the right. In order to eliminate the influence from Ω_4, we now define

$$(\tilde{S}_{14})_{11} = (S_{E_1E_4}p_1 + S_{E_1E_{10}}p_1)(x_k^H - h, y_k^H) \quad \left(\equiv (R_{E_1}A^{(1)}E^h\mathbf{p}_4)_1\right),$$

where we recall that $A^{(1)}$ is the local stiffness matrix on Ω_1. The last equality comes from the definition of the local Schur complement, and can be extracted from the subdomain solves used to construct the edge approximations.

Analogously, we define the seven remaining non-zero entries by:

$$\begin{aligned}
(\tilde{S}_{13})_{11} &\equiv (R_{E_1}A^{(4)}E^h\mathbf{p}_4)_1 \\
(\tilde{S}_{24})_{11} &\equiv (R_{E_2}A^{(2)}E^h\mathbf{p}_4)_1 \\
(\tilde{S}_{23})_{11} &\equiv (R_{E_2}A^{(3)}E^h\mathbf{p}_4)_1 \\
(\tilde{S}_{31})_{11} &\equiv (R_{E_3}A^{(4)}E^h\mathbf{p}_1)_1 \\
(\tilde{S}_{32})_{11} &\equiv (R_{E_3}A^{(3)}E^h\mathbf{p}_1)_1 \\
(\tilde{S}_{41})_{11} &\equiv (R_{E_4}A^{(1)}E^h\mathbf{p}_1)_1 \\
(\tilde{S}_{42})_{11} &\equiv (R_{E_4}A^{(2)}E^h\mathbf{p}_1)_1.
\end{aligned} \tag{25}$$

Symmetrization of $\tilde{S}_{V_k}$. Finally, in order to obtain a symmetric vertex approximation $\tilde{S}^P_{V_k}$ we use the *minimum-modulus* procedure:

$$(\tilde{S}^P_{V_k})_{ij} \equiv \begin{cases} (\tilde{S}_{V_k})_{ij} & \text{if } |(\tilde{S}_{V_k})_{ij}| \leq |(\tilde{S}_{V_k})_{ji}| \\ (\tilde{S}_{V_k})_{ji} & \text{if } |(\tilde{S}_{V_k})_{ji}| \leq |(\tilde{S}_{V_k})_{ij}|. \end{cases} \tag{26}$$

THEOREM 4.5. *The vertex matrix approximations $\tilde{S}^P_{V_k}$ are non-singular, diagonally dominant M-matrices.*

Proof. First, we note that since the fifth block row of $\tilde{S}^P_{V_k}$, is identical to the fifth block row of S_{V_k}, it has zero row sum. For any other row of $\tilde{S}^P_{V_k}$ centered about nodes not adjacent to the cross-point, the non-zero entries are the non-zero entries of the diagonal blocks $\tilde{S}_{ii}$, for $i = 1, 2, 3, 4$. These diagonal blocks were chosen as submatrices of $\tilde{S}^P_{E_1}$, $\tilde{S}^P_{E_2}$, $\tilde{S}^P_{E_3}$, and $\tilde{S}^P_{E_4}$, respectively, which were shown to be diagonally dominant M-matrices in Theorem 4.4, and therefore these rows are more diagonally dominant than the corresponding rows of S.

We now prove the diagonal dominance of the rows centered about nodes adjacent to the cross-point (x^H_k, y^H_k). Consider, for instance, the row sum corresponding to node $(x^H_k - h, y^H_k)$ to the left of the cross-point (x^H_k, y^H_k). The non-zero entries of this row are $(\tilde{S}_{11})_{11}$, $(\tilde{S}_{11})_{12}$, $(\tilde{S}_{13})_{11}$, $(\tilde{S}_{14})_{11}$, and $(\tilde{S}_{15})_{11}$. By construction:

$$\begin{array}{lcll} (\tilde{S}_{11})_{11} & = & \sum_{mod(j-1,3)=0}(S_{E_1} + S_{E_1E_6} + S_{E_1E_7})_{1,j} & > 0, \\ (\tilde{S}_{11})_{12} & = & \sum_{mod(j-2,3)=0}(S_{E_1} + S_{E_1E_6} + S_{E_1E_7})_{1,j} & \leq 0, \\ (\tilde{S}_{13})_{11} & = & \sum_{mod(j-1,3)=0}(S_{E_1E_3} + S_{E_1E_{10}})_{1,j} & \leq 0, \\ (\tilde{S}_{14})_{11} & = & \sum_{mod(j-1,3)=0}(S_{E_1E_4} + S_{E_1E_{13}})_{1,j} & \leq 0, \\ (\tilde{S}_{15})_{11} & = & (S_{E_1E_5})_{11} & \leq 0. \end{array}$$

By summing all these non-zero entries, we obtain

$$\begin{array}{lcl} \sum_j (\tilde{S}_{V_k})_{1j} & = & \sum_{mod(j-1,3)=0}(S_{E_1} + S_{E_1E_6} + S_{E_1E_7})_{1,j} + \\ & & \sum_{mod(j-2,3)=0}(S_{E_1} + S_{E_1E_6} + S_{E_1E_7})_{1,j} + \\ & & \sum_{mod(j-1,3)=0}(S_{E_1E_3} + S_{E_1E_{10}})_{1,j} + \\ & & \sum_{mod(j-1,3)=0}(S_{E_1E_4} + S_{E_1E_{13}})_{1,j} + \\ & & (S_{E_1E_5})_{11}. \end{array}$$

Since the right hand side is a *subset* of the corresponding row of S, which is strictly diagonally dominant, this shows that this row of $\tilde{S}^P_{V_k}$ is diagonally dominant. The proof of the diagonal dominance of the other rows centered about the nodes adjacent to (x^H_k, y^H_k) is analogous. Thus $\tilde{S}^P_{V_k}$ is strictly diagonally dominant in all rows except the one corresponding to the cross-point. This last property, together with the fact that $\tilde{S}^P_{V_k}$ has positive diagonal elements and non-positive off-diagonal elements, implies that $\tilde{S}^P_{V_k}$ is a non-singular M-matrix. □

4.2.3. Probe based preconditioner. We now define the Probe vertex space preconditioner (PVS) by

$$M^{-1}_{PVS} \equiv R^T_H A^{-1}_H R_H + \sum_{ij} R^T_{E_{ij}} (\tilde{S}^P_{E_{ij}})^{-1} R_{E_{ij}} + \sum_k R^T_{V_k} (\tilde{S}^P_{V_k})^{-1} R_{V_k}, \tag{27}$$

and the Probe BPS preconditioner (PBPS) by:

$$M^{-1}_{PBPS} \equiv R^T_H A^{-1}_H R_H + \sum_{ij} R^T_{E_{ij}} (\tilde{S}^P_{E_{ij}})^{-1} R_{E_{ij}}. \tag{28}$$

FIG. 5. *Discontinuos coefficients* $a(x,y)$

$a = 300$	$a = 10^{-4}$	$a = 31400$	$a = 5$
$a = 0.05$	$a = 6$	$a = 0.07$	$a = 2700$
$a = 10^{6}$	$a = 0.1$	$a = 200$	$a = 9$
$a = 1$	$a = 6000$	$a = 4$	$a = 140000$

5. Numerical Results. We now present results of numerical tests on the rate of convergence of the Fourier and Probe variants of the BPS and VS algorithms. The tests were conducted for the following elliptic problem:

$$\begin{cases} -\nabla \cdot (a(x,y)\nabla u) &= f & \text{in } \Omega = [0,1]^2 \\ u &= 0 & \text{on } \partial\Omega \end{cases}$$

for five choices of coefficients $a(x,y)$, various subdomain sizes H, and fine grid sizes h. The five coefficients used were:

1. $a(x,y) = I$, the Laplacian, see table 1.
2. $a(x,y) = I + 10(x^2+y^2)I$, slowly varying smooth coefficients, see table 2.
3. $a(x,y) = e^{10xy}I$, highly varying smooth coefficients, see table 3.
4. $a(x,y) = diag(1,\epsilon)$, anisotropic coefficients, see table 4.
5. Highly discontinuos coefficients of Fig. 5, see table 5.

The elliptic problem was discretized using the standard five-point difference stencil, see [29], on an $(n+1)\times(n+1)$ uniform fine grid with mesh size $h = 1/n$. The subdomains were chosen to be the sub-rectangles of an $(n_s+1)\times(n_s+1)$ uniform coarse grid with mesh size $H = 1/n_s$. Each subdomain, therefore consisted of $(n/n_s-1)\times(n/n_s-1)$ interior nodes. The coarse grid matrix A_H was chosen to be the five-point difference approximation of the elliptic problem on the coarse grid.

The entries of the exact solution were chosen randomly from the uniform distribution on $[-1,1]$ and the initial guess in the conjugate gradient method was chosen to be zero. The estimated condition number, κ, of the preconditioned system, and the number of iterations, ITN, required to reduce the initial residual by a factor of 10^{-5} (i.e., $\|r_k\|_2/\|r_0\|_2 \leq 10^{-5}$) are listed in the tables. During each iteration, the coarse grid problem and the subdomain problems were solved to high precision using a diagonally scaled preconditioned conjugate gradient method. The eigenvalues μ_k in the edge approximations $\tilde{S}^F_{E_{ij}}$ of (16) were chosen to be the Bramble, Pasciak and Schatz eigenvalues listed in (15), while the eigenvalues of the submatrices M_i^k of (18) were

TABLE 1
Laplace's equation: $a(x,y) = I$

h^{-1}	Ovlp	FBPS		PBPS		EVS		FVS		PVS	
H^{-1}	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_2	1/16	14.3	11	9.9	9	3.4	7	5.7	11	3.2	8
32_4	1/ 8	10.0	14	7.4	11	2.6	8	4.5	11	2.5	8
32_8	1/ 4	6.4	12	5.4	11	2.5	8	3.5	10	2.4	8
64_2	1/32	19.3	12	17.1	11	4.3	7	7.2	11	4.0	9
64_4	1/16	14.5	14	11.3	12	3.4	9	5.9	13	3.2	9
64_8	1/ 8	10.3	14	8.0	12	2.8	9	4.6	12	2.7	9
64_16	1/4	6.5	13	5.6	11	2.6	8	3.6	10	2.5	8
128_2	1/64	25.0	13	31.2	13	5.5	8	9.0	11	6.5	11
128_4	1/32	19.8	16	18.4	15	4.4	10	7.4	13	4.1	10
128_8	1/16	14.7	16	12.1	13	3.5	9	5.9	13	3.4	9
128_16	1/ 8	10.4	14	8.3	13	2.8	9	4.6	11	2.7	9
128_32	1/ 4	6.5	13	5.6	11	2.6	8	3.6	10	2.5	8
256_2	1/128	31.5	13	55.9	17	6.8	9	11.0	13	11.6	13
256_4	1/64	25.4	16	33.0	19	5.5	10	9.1	13	7.2	13
256_8	1/32	19.7	16	18.5	15	4.5	10	7.3	13	4.3	10
256_16	1/16	14.7	16	12.4	13	3.5	9	5.9	13	3.3	9
256_32	1/ 8	10.4	14	8.4	13	2.8	9	4.6	11	2.7	9
256_64	1/ 4	6.5	13	5.7	11	2.6	8	3.6	10	2.4	8

chosen to be the Dryja eigenvalues in (15). The Fourier and Probe BPS versions are denoted by FBPS and PBPS respectively, while the Fourier and Probe versions of the VS algorithms are denoted FVS and PVS, respectively. Unless otherwise stated, the number of nodes of overlap, N_{vs}, in the vertex regions is 1, i.e., there is one node on each vertex segment $V_k \cap E_{ij}$. The overlap ratio $\beta = h/H$ is listed as *Ovlp*.

Discussion. Tables 1 through 5 compares the performance of the various methods for the five sets of coefficients listed above. Table 1 corresponds to the Laplacian. In this case, the exact version of the VS algorithm, denoted by EVS, was also tested, because the eigenvalues of edge matrices $S_{E_{ij}}$ can be computed inexpensively using analytical formulas, see M_{Chan} in (15). In agreement with the theory, these results indicate that the Fourier variant FVS, has an observed rate of convergence independent of the mesh parameters H, h for fixed overlap ratio *Ovlp*. Moreover, the actual iteration numbers are quite insensitive to the choice of parameters H, h and *Ovlp*. For the range of subdomain and fine grid sizes tested, the performance of PVS is very similar to EVS. However, as the number of nodes per edge increases significantly, it is expected that the PVS version would deteriorate, based on properties of the probe preconditioner for two subdomains in [12]. The condition numbers for the variants of the BPS algorithms grow mildly with H/h, in agreement with theory. In most cases, due to clustering of eigenvalues of the preconditioned system, the number of iterations, ITN, was often better than that predicted by the condition numbers.

Tables 2 and 3 correspond to smoothly varying coefficients. Here again, the results are similar to those for the Laplacian, and are in agreement with the theory. Moreover, the rate of convergence of most variants are quite insensitive to the variations in the

TABLE 2
Mildly varying coefficients: $a(x,y) = \left(1 + 10(x^2+y^2)\right) I$

h^{-1}	Ovlp	FBPS		PBPS		FVS		PVS	
H^{-1}	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_2	1/16	15.2	11	10.6	9	6.0	11	3.4	8
32_4	1/ 8	10.2	14	7.6	11	4.6	11	2.6	8
32_8	1/ 4	6.4	12	5.4	11	3.6	10	2.4	8
64_2	1/32	20.4	12	17.8	11	7.5	11	4.2	9
64_4	1/16	14.9	14	11.6	12	5.8	12	3.2	9
64_8	1/ 8	10.3	14	8.1	12	4.6	11	2.7	9
64_16	1/ 4	6.5	13	5.6	11	3.6	10	2.4	8
128_2	1/64	26.3	13	32.1	13	9.4	11	6.7	11
128_4	1/32	20.0	16	18.4	15	7.3	13	4.2	10
128_8	1/16	14.7	16	12.2	13	5.9	13	3.4	9
128_16	1/ 8	10.4	14	8.4	13	4.6	11	2.7	8
128_32	1/ 4	6.5	13	5.6	11	3.6	10	2.4	8
256_2	1/128	32.9	13	57.0	16	11.5	13	11.7	13
256_4	1/64	25.8	17	33.2	19	9.3	13	7.2	13
256_8	1/32	19.9	16	18.6	15	7.3	13	4.3	10
256_16	1/16	14.7	16	12.3	13	5.9	13	3.4	9
256_32	1/ 8	10.4	14	8.4	13	4.6	11	2.7	9
256_64	1/ 4	6.5	13	5.7	11	3.6	10	2.4	8

coefficients $a(x,y)$. In order to see the importance of scalings, in table 3 we also tested a variant nsFVS of the FVS preconditioner, in which the edge approximations were not diagonally scaled, but were instead scaled by a scalar α_{ij} on each edge E_{ij}, i.e.

$$\tilde{S}^F_{E_{ij}} \equiv \alpha_{ij} W diag(\mu_k) W,$$

where

$$\alpha_{ij} \equiv \frac{a(x_i,y_i) + a(x_j,y_j)}{2},$$

for some point $(x_i,y_i) \in \Omega_i$ and $(x_j,y_j) \in \Omega_j$. As the results indicate, this variant was sensitive to the variations in the coefficients.

Table 4 concerns the case of anisotropic coefficents. Here, the results are qualitatively different from the preceeding cases. Note that the rate of convergence of all variants of the VS and BPS algorithms deteriorate to a fixed rate as $\epsilon \to 0$. The limiting condition numbers seem to depend on the coarse mesh size, as $1/H$. A possible explanation for this deterioration is the following. For $\epsilon = 0$, the unknowns are essentially coupled only along the x axis and adjacent vertical edges are coupled strongly in the Schur complement. This coupling is not represented in the VS preconditioner, and may cause the deterioration in the convergence rate. The results in table 4 also indicate that the probe versions perform slightly better than the Fourier versions. This can be explained as follows. For $\epsilon = 0$, the edge matrices $S_{E_{ij}}$ on the horizontal edges become a discrete approximation of $-d^2/dx^2$, while on vertical edges $S_{E_{ij}}$ becomes a nearly diagonal matrix, similar to the identity. The FVS edge matrices $\tilde{S}^P_{E_{ij}}$ approximate the square root of the Laplacian, and are therefore invalid in this case. By

TABLE 3
Highly varying coefficients: $a(x,y) = e^{10xy}I$

h^{-1}	Ovlp	FBPS		PBPS		nsFVS		FVS		PVS	
H^{-1}	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_2	1/16	22.5	11	18.4	9	16.1	18	7.5	11	4.4	9
32_4	1/ 8	13.4	15	11.0	13	7.2	13	5.1	11	3.2	9
32_8	1/ 4	7.0	12	6.2	11	4.0	10	3.9	10	2.5	8
64_2	1/32	28.9	12	25.9	11	24.5	23	9.5	11	5.8	9
64_4	1/16	17.6	16	15.5	15	11.3	16	6.5	12	4.0	9
64_8	1/ 8	11.0	12	9.1	12	5.6	12	4.9	11	2.8	8
64_16	1/ 4	6.6	12	5.8	11	3.7	10	3.7	10	2.5	8
128_2	1/64	36.3	13	45.0	14	35.8	28	11.8	12	8.6	11
128_4	1/32	24.4	16	23.3	15	16.1	19	8.4	13	5.1	10
128_8	1/16	15.7	14	13.2	13	7.7	14	6.0	12	3.6	10
128_16	1/ 8	10.4	14	8.4	11	4.7	12	4.6	11	2.8	9
128_32	1/ 4	6.5	12	5.7	11	3.6	10	3.6	10	2.4	8
256_2	1/128	44.2	14	77.2	17	32.0	24	14.4	13	15.1	14
256_4	1/64	29.3	17	41.4	22	16.2	19	10.1	13	8.5	13
256_8	1/32	20.8	16	20.2	15	8.0	14	7.7	13	4.4	10
256_16	1/16	15.0	15	12.4	13	5.0	11	6.1	13	3.3	9
256_32	1/ 8	10.3	14	8.2	12	3.8	10	4.7	12	2.7	8
256_64	1/ 4	6.5	12	5.6	11	2.9	9	3.6	10	2.4	8

construction, the tridiagonal probing technique approximates diagonal and tridiagonal matrices well, and consequently, they perform better than the Fourier versions we tested. The algorithms for anisotropic problems need further study.

Table 5 refers to the case of the highly discontinuous coefficients of Fig. 5. The performance is similar to the case of smooth coefficients, and the results indicate that the rate of convergence of all variants is quite insensitive to the jumps in the coefficients.

In tables 6 and 7, we compare various preconditioners for different choices of eigenvalues μ_k in the Fourier approximations (16). Here, CFBPS denotes that the eigenvalues of the Fourier edge approximations in the FBPS preconditioner were those of M_{Chan} in (15), while CFVS denotes that the same eigenvalues were used in the FVS preconditioner. In agreement with theory, the Fourier versions were spectrally equivalent with respect to variations in H and h, for fixed overlap $Ovlp$. Amongst the various eigenvalues tested, the exact eigenvalues of the Schur complement of the Laplacian used in CFBPS and CFVS gave the best results. Corresponding rates for the probe version are also listed for comparison.

Finally, in tables 8, 9, 10, and 11, we present a comparison of the FVS and PVS preconditioners, as the amount of overlap N_{VS} in the vertex regions is increased. Here, $N_{VS} = 0$ indicates that only the vertex node was used, i.e., the vertex matrices were 1×1. We note that the improvement in condition number of the VS algorithms as the overlap $Ovlp$ is increased is mild, as also noted in [27]. In particular, the performance is quite satisfactory even when the vertex region consists of just one point, see Widlund [31].

TABLE 4
Anisotropic problem: $\frac{\partial^2 u}{\partial x^2} + \epsilon \frac{\partial^2 u}{\partial y^2} = f$

$h^{-1}\_H^{-1}$	64_2				64_4				64_16			
	PVS		FVS		PVS		FVS		PVS		FVS	
ϵ	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN
0.1	7.4	10	14.5	17	5.9	12	12.0	18	9.0	16	12.9	19
0.08	8.0	10	16.1	17	6.4	12	13.5	20	10.8	17	15.5	20
0.06	8.9	10	18.4	19	7.5	13	15.2	21	13.6	18	19.7	23
0.04	10.3	10	22.4	21	9.7	14	20.9	24	19.2	22	27.9	26
0.02	13.0	10	31.4	24	13.0	16	29.4	28	34.4	27	50.1	33
0.01	16.3	10	43.6	27	20.7	18	41.7	31	58.4	34	84.8	41
10^{-3}	29.3	8	115.6	38	60.3	25	151.5	47	215.8	59	351.8	73
10^{-4}	39.4	7	179.8	46	81.7	25	250.7	57	352.5	69	591.4	92
10^{-5}	41.8	6	193.8	48	105.1	27	253.6	59	396.6	73	583.6	87
10^{-6}	42.0	6	195.3	49	105.0	26	267.7	59	355.8	71	647.5	93
10^{-7}	42.1	6	195.4	48	102.1	25	273.1	59	405.6	73	654.0	92
10^{-8}	42.1	6	195.2	48	106.2	23	254.9	57	395.6	72	661.7	93

TABLE 5
Discontinuous coefficients: See $a(x, y)$ *of Fig. 5.*

h^{-1}	Ovlp	FBPS		PBPS		FVS		PVS	
$\_H^{-1}$	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_4	1/ 8	10.2	13	7.5	11	6.1	12	8.1	11
32_8	1/ 4	6.6	12	5.2	10	8.5	13	3.7	9
64_4	1/16	14.7	15	11.1	11	9.3	14	10.1	11
64_8	1/ 8	10.1	14	8.1	12	8.4	14	5.2	10
64_16	1/ 4	6.5	13	5.6	11	6.9	12	4.1	9
128_4	1/32	19.6	17	18.1	16	12.3	14	6.8	11
128_8	1/16	14.4	16	12.1	14	11.5	15	5.9	11
128_16	1/ 8	10.2	14	8.3	13	6.4	13	3.4	9
128_32	1/ 4	6.6	13	5.7	11	6.8	12	4.1	9
256_4	1/64	25.4	19	33.0	17	14.9	15	7.8	13
256_8	1/32	19.3	17	18.7	16	8.8	15	4.9	11
256_16	1/16	14.8	16	12.3	13	12.4	16	6.9	11
256_32	1/ 8	10.3	14	8.4	13	8.6	14	6.0	10
256_64	1/ 4	6.5	13	5.7	11	6.0	12	4.1	9

TABLE 6
Different Edge Fourier Preconditioners for Laplace Equation

h^{-1}	Ovlp	FBPS		CFBPS		FVS		CFVS		PVS	
H^{-1}	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_2	1/16	14.3	11	9.5	7	5.7	11	4.6	8	3.2	8
32_4	1/ 8	10.0	14	7.3	11	4.5	11	3.6	9	2.5	8
32_8	1/ 4	6.4	12	5.3	11	3.5	10	2.9	9	2.4	8
64_2	1/32	19.3	12	13.4	7	7.2	11	5.8	8	4.0	9
64_4	1/16	14.5	14	10.7	11	5.9	13	4.7	10	3.2	9
64_8	1/ 8	10.3	14	8.1	12	4.6	12	3.7	10	2.7	9
64_16	1/ 4	6.5	13	5.5	11	3.6	10	2.9	9	2.5	8
128_2	1/64	25.0	13	17.8	8	9.0	11	7.3	8	6.5	11
128_4	1/32	19.8	16	14.6	12	7.4	13	5.8	10	4.1	10
128_8	1/16	14.7	16	11.5	14	5.9	13	4.7	10	3.4	9
128_16	1/ 8	10.4	14	8.3	13	4.6	11	3.7	10	2.7	9
128_32	1/ 4	6.5	13	5.5	11	3.6	10	2.9	9	2.5	8
256_2	1/128	31.5	13	23.0	7	11.0	12	8.9	9	11.6	13
256_4	1/64	25.4	16	19.2	13	9.0	14	7.3	10	7.2	13
256_8	1/32	19.7	16	15.6	13	7.2	13	5.9	11	4.3	10
256_16	1/16	14.7	16	11.7	14	5.9	13	4.7	10	3.3	9
256_32	1/ 8	10.4	14	8.4	13	4.6	11	3.8	10	2.7	9
256_64	1/ 4	6.5	13	5.5	11	3.6	10	2.9	9	2.5	8

TABLE 7
Different Edge Fourier Preconditioners for $a(x,y) = e^{10xy}I$

h^{-1}	Ovlp	FBPS		CFBPS		FVS		CFVS		PVS	
H^{-1}	h/H	κ	ITN	κ	ITN	κ	ITN	κ	ITN	κ	ITN
32_2	1/16	22.5	11	18.1	8	7.5	11	6.2	9	4.4	9
32_4	1/ 8	13.4	15	10.7	13	5.1	11	4.5	10	3.2	9
32_8	1/ 4	7.0	12	5.8	11	3.9	10	3.3	9	2.5	8
64_2	1/32	28.9	12	23.0	9	9.5	11	7.7	9	5.8	9
64_4	1/16	17.6	16	14.7	12	6.5	12	5.4	9	4.0	9
64_8	1/ 8	11.0	12	8.8	11	4.9	11	4.0	10	2.8	8
64_16	1/ 4	6.6	12	5.6	11	3.7	10	3.0	9	2.5	8
128_2	1/64	36.3	13	28.5	9	11.8	12	9.6	9	8.6	11
128_4	1/32	24.4	16	19.4	13	8.4	13	7.0	9	5.1	10
128_8	1/16	15.7	14	12.5	11	6.0	12	5.1	10	3.6	10
128_16	1/ 8	10.4	14	8.5	12	4.6	11	3.8	9	2.8	9
128_32	1/ 4	6.5	12	5.5	11	3.6	10	3.0	9	2.4	8
256_2	1/128	44.2	14	34.7	9	14.4	13	11.6	9	15.1	14
256_4	1/64	29.3	17	23.3	14	10.1	13	8.3	10	8.5	13
256_8	1/32	20.8	16	16.5	13	7.7	13	6.2	10	4.4	10
256_16	1/16	15.0	15	11.9	12	6.1	13	4.8	10	3.3	9
256_32	1/ 8	10.3	14	8.3	12	4.7	12	3.8	10	2.7	8
256_64	1/ 4	6.5	12	5.4	11	3.6	10	2.9	9	2.4	8

TABLE 8
Variation of vertex sizes for $H = 1/2$, $h = 1/128$, and $a(x, y) = I$.

N_{vs}	0	1	2	3	4	5	6	7
κ_{FVS}	7.45	8.97	8.07	7.66	6.85	6.98	6.71	6.53
ITN	10	11	12	12	12	13	12	12

TABLE 9
Variation of vertex sizes for $H = 1/2$, $h = 1/128$, and $a(x, y) = e^{10xy} I$.

vs	0	1	2	3	4	5	6	7
κ_{FVS}	9.85	11.80	10.25	10.00	9.41	9.01	8.63	8.40
ITN	11	12	12	13	12	12	12	13

TABLE 10
Variation of vertex sizes for $H = 1/2$, $h = 1/128$, $a(x, y) = I$.

N_{vs}	0	1	2	3	4	5	6	7
κ_{PVS}	8.3	6.6	5.6	5.0	4.8	3.2	4.6	4.5
ITN	11	11	11	11	11	9	11	11

TABLE 11
Variation of vertex sizes for $H = 1/2$, $h = 1/128$ and $a(x, y) = e^{10xy} I$.

vs	0	1	2	3	4	5	6	7
κ_{PVS}	10.8	9.1	7.3	6.6	6.6	6.6	6.8	6.9
ITN	12	11	10	10	10	11	11	11

Conclusions: Both the Fourier and Probe variants of the vertex space algorithm are designed to be efficient alternatives to the original VS algorithm. Our experiments for a wide range of coefficients and grid sizes show that the efficiency does not come at a price of deteriorated performance. We hope that these variants will provide flexible and efficient methods for solving second order elliptic problems using the domain decomposition approach.

REFERENCES

[1] V. I. AGOSHKOV, *Poincaré-Steklov operators and domain decomposition methods in finite dimensional spaces*, in First International Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, G. H. Golub, G. A. Meurant, and J. Périaux, eds., Philadelphia, 1988, SIAM.

[2] P. E. BJØRSTAD AND O. B. WIDLUND, *Iterative methods for the solution of elliptic problems on regions partitioned into substructures*, SIAM J. Numer. Anal., 23 (1986), pp. 1093 – 1120.

[3] J. H. BRAMBLE, J. E. PASCIAK, AND A. H. SCHATZ, *The construction of preconditioners for elliptic problems by substructuring, I*, Math. Comp., 47 (1986), pp. 103– 134.

[4] ———, *An iterative method for elliptic problems on regions partitioned into substructures*, Math. Comp., 46 (1986), pp. 361–369.

[5] T. CHAN, R. GLOWINSKI, , J. PÉRIAUX, AND O. WIDLUND, eds., *Domain Decomposition Methods*, Philadelphia, 1989, SIAM. Proceedings of the Second International Symposium on Domain Decomposition Methods , Los Angeles, California , January 14 - 16, 1988.

[6] ———, eds., *Domain Decomposition Methods*, Philadelphia, 1989, SIAM. Proceedings of the Third International Symposium on Domain Decomposition Methods , Houston, Texas, 1989.

[7] ———, eds., *Domain Decomposition Methods*, Philadelphia, 1989, SIAM. Proceedings of the Fourth International Symposium on Domain Decomposition Methods , Moscow, USSR, 1990.

[8] T. F. CHAN, *Analysis of preconditioners for domain decomposition*, SIAM J. Numer. Anal., 24 (1987), pp. 382–390.

[9] T. F. CHAN AND D. GOOVAERTS, *A note on the efficiency of domain decomposed incomplete factorizations*, SIAM J. Sci. Stat. Comput., 11 (July 1990), pp. 794 – 803.

[10] T. F. CHAN AND T. Y. HOU, *Domain decomposition interface preconditioners for general 2nd order elliptic problems*, Tech. Rep. CAM 88-16, Department of Mathematics, UCLA, 1990. To appear in SIAM Journal of Scientific and Statistical Computing, 1992.

[11] T. F. CHAN AND T. P. MATHEW, *An application of the probing technique to the vertex space method in domain decomposition*, in Domain Decomposition Methods for Partial Differential Equations, T. F. Chan, R. Glowinski, J. Periaux, and O. Widlund, eds., SIAM, 1990.

[12] ———, *The interface probing technique in domain decomposition*, Siam J. Matrix Analysis and Applications, 13 (1992).

[13] T. F. CHAN AND D. C. RESASCO, *A survey of preconditioners for domain decomposition*, Tech. Rep. /DCS/RR-414, Yale University, 1985.

[14] P. G. CIARLET, *The Finite Element Method for Elliptic Problems*, North-Holland, 1978.

[15] A. R. CURTIS, M. J. POWELL, AND J. K. REID, *On the estimation of sparse Jacobian matrices*, J. Inst. Maths. Applics., 13 (1974), pp. 117–120.

[16] M. DRYJA, *A capacitance matrix method for Dirichlet problem on polygon region*, Numer. Math., 39 (1982), pp. 51 – 64.

[17] M. DRYJA AND O. B. WIDLUND, *Some domain decomposition algorithms for elliptic problems*, in Proceedings of the Conference on Iterative Methods for Large Linear Systems held in Austin, Texas, October 1988, to celebrate the Sixty-fifth Birthday of David M. Young, Jr., Academic Press, Orlando, Florida, 1989., 1989.

[18] S. C. EISENSTAT, Personal Communication, 1985.

[19] D. FUNARO, A. QUARTERONI, AND P. ZANOLLI, *An iterative procedure with interface relaxation for domain decomposition methods*, SIAM J. Numer. Anal., 25 (1988), pp. 1213 –1236.

[20] R. GLOWINSKI, G. H. GOLUB, G. A. MEURANT, AND J. PÉRIAUX, eds., *Domain Decomposition Methods for Partial Differential Equations*, Philadelphia, 1988, SIAM. Proceedings of the First International Symposium on Domain Decomposition Methods for Partial Differential Equations, Paris, France, January 1987.

[21] G. GOLUB AND D. MAYERS, *The use of preconditioning over irregular regions*, in Computing Methods in Applied Sciences and Engineering, VI, R. Glowinski and J. L. Lions, eds., Amsterdam, New York, Oxford, 1984, North-Holland, pp. 3–14. Proceedings of a conference held in Versailles, France, December 12-16,1983.

[22] G. H. GOLUB AND C. F. V. LOAN, *Matrix Computations*, Johns Hopkins Univ. Press, 1989. Second Edition.

[23] D. E. KEYES AND W. D. GROPP, *A comparison of domain decomposition techniques for elliptic partial differential equations and their parallel implementation*, SIAM J. Sci. Stat. Comput., 8 (1987), pp. s166 – s202.

[24] ———, *Domain decomposition techniques for the parallel solution of nonsymmetric systems of elliptic bvps*, in Domain Decomposition Methods, T. Chan, R. Glowinski, J. Périaux, and O. Widlund, eds., Philadelphia, 1989, SIAM.

[25] S. V. NEPOMNYASCHIKH, *On the application of the method of bordering for elliptic mixed boundary value problems and on the difference norms of* $W_2^{1/2}(S)$, Tech. Rep. 106, Computing Center of the Siberian Branch of the USSR Academy of Sciences, Novosibirsk, 1984. In Russian.

[26] M. J. POWELL AND P. L. TOINT, *On the estimation of sparse Hessian matrices*, SIAM J. Numerical Analysis, 16 (1979), pp. 1060–1074.

[27] B. F. SMITH, *An optimal domain decomposition preconditioner for the finite element solution of linear elasticity problems*, Tech. Rep. 482, Department of Computer Science, Courant Institute, 1989. To Appear in Proceedings of Copper Mountain Conference on Iterative Methods, SIAM Journal of Scientific and Statistical Computing.

[28] ———, *Domain Decomposition Algorithms for the Partial Differential Equations of Linear Elasticity*, PhD thesis, Courant Institute, New York, N.Y., 1990.

[29] R. S. VARGA, *Matrix Iterative Analysis*, Prentice-Hall, 1962.

[30] O. B. WIDLUND, *Iterative substructuring methods: Algorithms and theory for elliptic problems in the plane*, in First International Symposium on Domain Decomposition Methods for Partial Differential Equations, R. Glowinski, G. H. Golub, G. A. Meurant, and J. Périaux, eds., Philadelphia, 1988, SIAM.

[31] ———, *Some schwarz methods for symmetric and non-symmetric elliptic problems*, Tech. Rep. 0581, Courant Institute, New York University, 1991.